

Lecture 05

Java – Why Java

Mostafa Elhosseini

Professor at College of Computer Science and
Engineering

Taibah University

<https://youtube.com/drmelhosseini>

| Java

- Sun Microsystems in 1991 funded an internal corporate research project led by **James Gosling**, which resulted in a **C++-based** object-oriented programming language that **Sun called Java**.
- Using Java, you can write programs that will run on a great variety of computer systems and computer-controlled devices.
- This is sometimes called “**write once, run anywhere.**”

| Java

- world's most widely used computer programming languages and, according to the **TIOBE** Index, the world's most popular
- **Java** is also widely used for implementing **Internet-based** applications and **software for devices that communicate** over a network
- There are now **10 million Java** developers worldwide and Java runs on 15 billion devices including **two billion vehicles** and **350 million medical** devices
- The explosive growth of mobile phones, tablets and other devices is creating significant opportunities for programming mobile apps

| Java

Access control systems

Automobiles

Cable boxes

CT scanners

Game consoles

Home security systems

Logic controllers

Mobile phones

Optical sensors

Point-of-sale terminals

Routers

Smart meters

Tablets

Transportation passes

Airplane systems

Blu-ray Disc™ players

Copiers

Desktop computers

GPS navigation systems

Internet-of-Things gateways

Lottery systems

MRIs

Parking meters

Printers

Servers

Smartpens

Televisions

TV set-top boxes

ATMs

Building controls

Credit cards

e-Readers

Home appliances

Light switches

Medical devices

Network switches

Personal computers

Robots

Smart cards

Smartphones

Thermostats

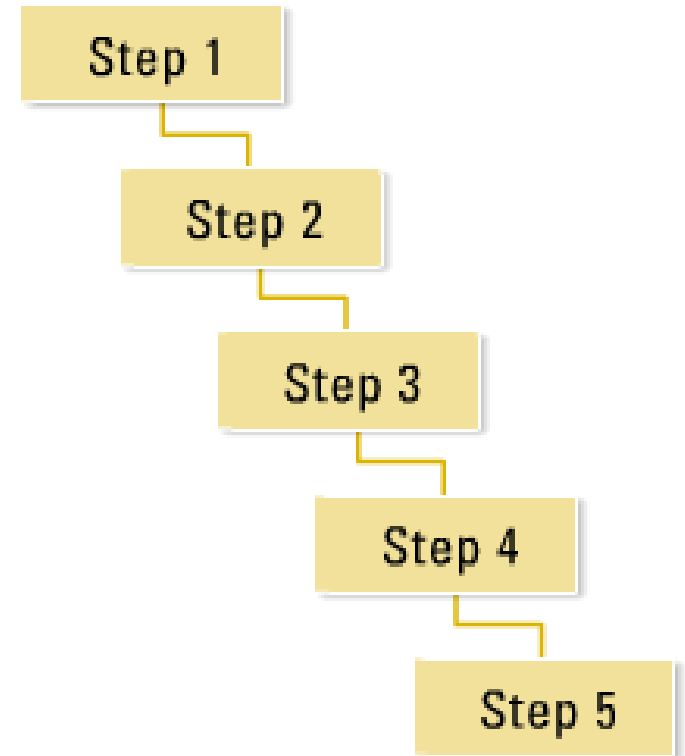
Vehicle diagnostic systems

| Why Java?

- The JAVA Programming language was designed to be
 - **Object-oriented**
 - Distributed
 - Simple
 - Multithreaded
 - Secure
 - Platform-independent

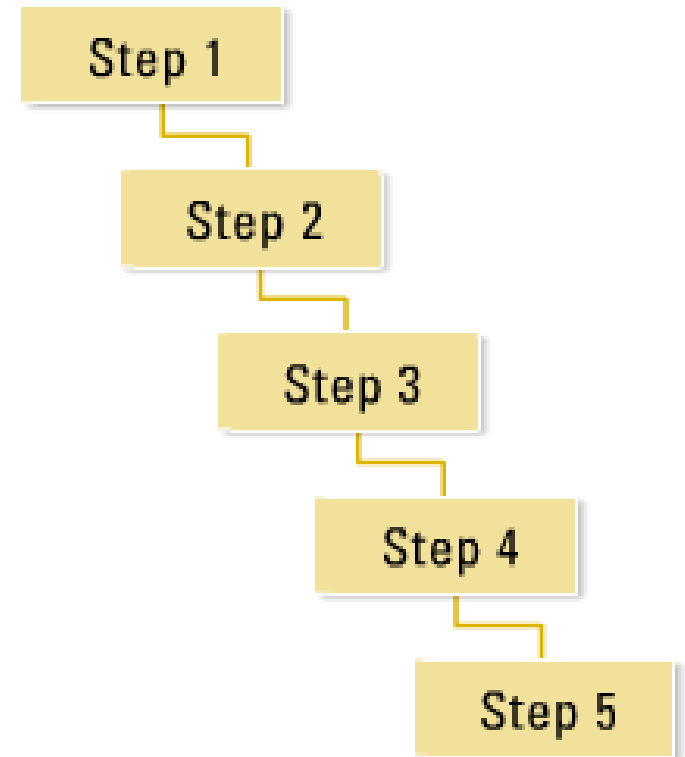
| Procedural Programming

- How programs in the past have been structured?
- Procedural program, a program is **broken up into a series of steps**, you perform **step1** and then the **next step**, and then the next step in a very linear fashion



| Procedural Programming

- Each statement in the language tells the computer to do something:
 - Get some input,
 - add these numbers,
 - divide by six,
 - display that output.
- Get up – have breakfast – go to work
- Procedural thinking is natural
- Works great if you want to plan your day – but for city?



| Real-world Modeling

- In the physical world we deal with objects such as **people** and **cars**.
- Complex real-world objects have both **attributes** and **behavior**.
 - **Attributes**: sometimes called **characteristics**
 - for people, **eye color** and **job title** and
 - for **cars**, **horsepower** and **number of doors**.

| Behavior

- Behavior is something a real-world object does in response to some stimulus.
 - If you ask your boss for a raise, she will generally say yes or no.
 - If you apply the brakes in a car, it will generally stop.
- **Behavior** is like a **function (method)** : you call a function to do something (display the inventory, for example) and it does it.

| What kinds of things become objects in object-oriented programs?

Physical Objects

- Electrical components
- Countries
- Aircraft

computer-user environment

- Windows
- Menus
- Graphics objects (lines, rectangles, circles)
- The mouse, keyboard, disk drives, printer

Data-storage constructs

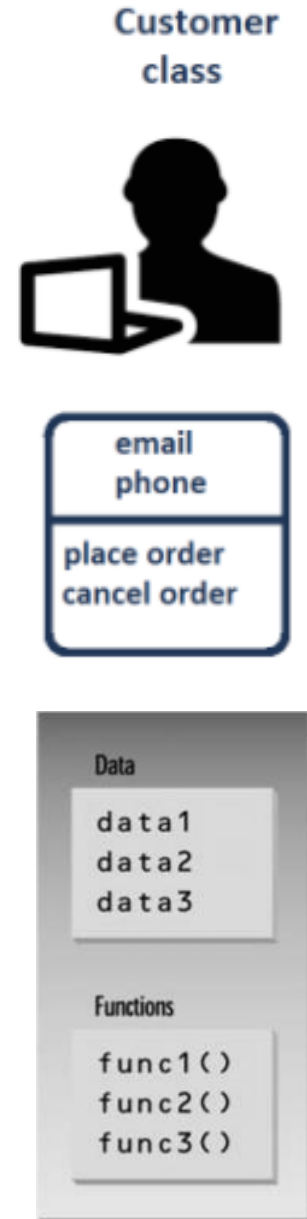
- Customized arrays
- Stacks
- Linked lists
- Binary trees

Human entities

- Employees
- Students
- Customers
- Salespeople

| Classes

- Let's look at an analogy. Almost all computer languages have built-in data types. For instance, a data type **int**, meaning integer, is **predefined in C++**
- You can **declare as many variables of type int as you need** in your program:
 - `int day;`
 - `int count;`
- In a similar way, you can **define many objects of the same class**



| Classes

- A class serves as a plan, or blueprint
- Every class is like a recipe which can be used when you want to create a useful object
- If you say every customer will have a phone number and email and will be able to place and cancel order – you just define a class
- A class is thus a description of a number of similar objects.
 - Mercedes-Benz, BMW, and Jaguar are members of the top luxury car brands class.
 - There is no one person called “top luxury car ,” but specific car with specific brands are members of this class if they possess certain characteristics.
- An object is often called an “instance” of a class.

| Object

- Object is a particular representation “realization” of a class, while a class is just an abstraction pattern
- Objects are incarnations of ideas expressed in classes, like a cheesecake on your plate is an incarnation of the idea expressed in a recipe printed in an old cookbook
- An object is a data structure incorporating information about state and behaviors
- There is no clear border between data and code: they live as one in objects.

| Object

- An object representing a customer can have a certain phone number and email associated with them, and behavior like **placeOrder** or **cancelOrder**
- An object representing a button on a website can have a label, and can trigger event when pressed
- The objects interact with each other, exchanging data or activating their methods.
- A properly constructed class (and thus, its objects) are able to protect the sensible data and hide it from unauthorized modifications.

| Merits of OOP

- Encapsulation
- Inheritance
- Reusability
- Polymorphism
- Overloading

| References

- Java: How to Program, 9th Edition (Deitel) 9th Edition by Paul Deitel , Harvey M. Deitel
- Head First Java, 2nd Edition 2nd Edition by Kathy Sierra, Bert Bates
- Think Java: How to Think Like a Computer Scientist 1st Edition by Allen B. Downey, Chris Mayfield